Journal of Algebraic Structures and Their Applications
ISSN: 2382-9761
www.as.yazd.ac.ir

Algebraic Structures and Their Applications Vol. 6 No. 1 (2019) pp 83-97.

ISOGENY-BASED CERTIFICATELESS IDENTIFICATION SCHEME

HASSAN DAGHIGH* AND RUHOLLA KHODAKARAMIAN GILAN

Abstract. In this paper, we propose a new certificateless identification scheme based on isogenies between elliptic curves that is a candidate for quantum-resistant problems. The proposed scheme has the batch verification property which allows verifying more than one identity by executing only a single challenge-response protocol.

1. Introduction

In 1978, Rivest, Shamir and Adleman [13] proposed the first public-key encryption scheme that allows an entity to securely send a message to another entity without sharing any secret key between two parties. The fundamental point of public-key encryption is generating a pair of keys instead of just one, a public key for encryption and a related private key for decryption.

Despite its many benefits, history has shown that public key encryption faces a significant practical problem: the sender has to be sure that the used public key is indeed the intended receiver's public key. Therefore, the presence of a trusted third party who can be relied upon to check a receiver's identity and guarantee the accuracy of the public key seems to be necessary. Hence the user has to bind his public key to his identity using a certificate obtained from a

DOI: 10.29252/as.2019.1357

MSC(2010): Primary:14H52, 94A60.

Keywords: Certificateless Identification Scheme, Elliptic Curves, Isogeny, Cryptography, Pairing.

Received: 30 July 2018, Accepted: 06 March 2019.

∗Corresponding author

certificate authority. Nevertheless, this method leads to the certificate management problem as the number of users increases.

The same problem occurs in identification schemes in which a prover tries to confirm his identity such that an intruder watching the information flow, cannot easily impersonate the prover's identity. In 1984, Shamir proposed the idea of identity-based cryptography to deal away with the problem of certificate management [14]. However, the first identity-based identification schemes were proposed by Bellare et al. [2] and Kurosawa [12] in 2004 independently. In these schemes, Bellare and Kurosawa replaced the public key infrastructure with a trusted authority to compute the private key for the users. This method seemed to be more efficient than the certificate-based scheme, but the disadvantage was that the trusted authority who had created the private key pairs had access to the user's private key in the system.

In 2003, Al-Riyami and Paterson introduced the notion of certificateless public key cryptography, that escaped the weakness of identity-based cryptography despite maintaining its attractive properties [1]. In this scheme, the idea is that the trusted third party called the Key Generation Center (KGC) generates a partial private key for each user. Then the user combines the private key with its own selected secret value to create his full private key. Therefore, the key generation center doesn't have access to the user's full private key. After presenting this approach, many certificateless encryption and identification schemes have appeared with many security assumptions in literature. Using the user-selected secret value in certificateless public key cryptography, not only removes the inherited key escrow property from the identity-based public key cryptosystem but also makes the user free from obtaining a certificate from the trusted authority to establish the authenticity of his public key.

In 2013, Chin et al. proposed the first model of certificateless identification scheme (CLI) in [5] which offered an alternative solution to the certificate management problem of traditional identification schemes. The security of many of the current identification schemes are based on the hardness of discrete logarithm problem. Since the appearance of quantum computers in the future may cause a serious threat to the security of these protocols, post-quantum cryptography searches to design cryptosystems that are secure against both quantum and classical computers simultaneously. Some of these quantum-resistant cryptosystems are lattice-based cryptosystems, code-based cryptosystems, McEliece cryptosystem and multivariate public key cryptography.

Isogeny-based cryptosystems seem to be a promising candidate for quantum-resistant cryptography. In 2011, Luca de Feo et al. proposed a new quantum-resistant zero-knowledge identification scheme using isogenies between supersingular elliptic curves and detailed security proofs for the protocols [9]. In this paper, we propose a new certificateless identification

scheme using isogenies between elliptic curves. This scheme is a challenge-response identification protocol using pairings on elliptic curves. Moreover, it has the batch verification property which means that by a single execution of the protocol, the verifier can check more than one identity at the same time. Developing a sub-exponential time quantum algorithm to break isogenies between ordinary elliptic curves by Childs et al. [4] motivates us to use supersingular elliptic curves for which the fastest known quantum attack remains exponential because of the non-commutativity of the endomorphism ring.

In this paper, we first briefly review the concept of elliptic curves, isogenies and some useful properties of these maps in section 2. In section 3, we present a certificateless identification scheme first proposed by Chin et al. [6]. Then we continue with constructing a certificateless identification scheme on the additive group of elliptic curves inspired by Chin et al. [6] scheme on multiplicative groups. In section 4, we present our certificateless identification scheme using isogenies between elliptic curves. This protocol can be applied specially on supersingular elliptic curves to increase the security against quantum attacks due to the existence of a subexponential attack for the isogeny problem for ordinary elliptic curves. Moreover, the proposed scheme has the batch verification property and can be used to verify more than one identity by executing a single challenge-response protocol.

2. Preliminaries

Elliptic Curves and Isogenies: In this section, we introduce some basic concepts in elliptic curves. For more details, one can refer to [15, 20].

Definition 2.1. Let $q = p^\alpha$, where p is a prime number and α is a positive integer. An elliptic curve E over the finite field $\mathbb{F}_q$ is a non-singular projective plane curve defined by the equation

$$E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6,\ a_1, a_2, a_3, a_4, a_6 \in \mathbb{F}_q$$

In characteristic $p \neq 2, 3$, this equation can be reduced to the short form

$$E : y^2 = x^3 + ax + b,\ a, b \in \mathbb{F}_q$$

Moreover, the set of $\mathbb{F}_q$-rational points of E is defined as

$$E(\mathbb{F}_q) = \{(x, y) \in E \mid x, y \in \mathbb{F}_q\} \cup \{\mathcal{O}\}.$$

where $\mathcal{O}$ is the point at infinity.

The set $E(\mathbb{F}_q)$ forms an abelian additive group with $\mathcal{O}$ as the trivial element. For a point $P \in E(\mathbb{F}_q)$, the order of P is the least positive integer n such that $nP = \mathcal{O}$ and it is denoted by $\mathrm{ord}(P)$.

In the following, we consider maps between elliptic curves that preserve the algebraic structure of the group of points of elliptic curves.

Definition 2.2. Let E_1 and E_2 be two elliptic curves defined over the finite field $\mathbb{F}_q$. An isogeny $\phi : E_1 \to E_2$, is an algebraic map which is a group homomorphism.

For every isogeny $\phi : E_1 \to E_2$, the degree of ϕ which is its degree as an algebraic map, is denoted by $deg(\phi)$. For separable isogenies, we have $deg(\phi) = |ker(\phi)|$. Two elliptic curves E_1 and E_2 are l-isogenous if there exists an isogeny of degree l from E_1 to E_2. For every l-isogeny $\phi : E_1 \to E_2$, there exists an l-isogeny $\widehat{\phi} : E_2 \to E_1$ such that $\phi\widehat{\phi} = [l]_{E_2}$ and $\widehat{\phi}\phi = [l]_{E_1}$, where $[l]_{E_i}$ is the multiplication-by-l map on E_i for $i = 1, 2$. The isogeny $\widehat{\phi}$ is called the dual of ϕ.

By Tate's theorem, two elliptic curves E_1 and E_2 are isogenous over the finite field $\mathbb{F}_q$ if and only if they have the same number of points over $\mathbb{F}_q$ [18].

For every subgroup G of E_1, there exists an elliptic curve E_2 (unique up to isomorphism) and an isogeny $\psi : E_1 \longrightarrow E_2$ with kernel G. This isogeny can be computed using Velu's formula or using kernel polynomial of the subgroup G [11].

The group of all isogenies from E_1 to E_2 is denoted by $Hom(E_1, E_2)$ and $End(E) = Hom(E, E)$ denotes the endomorphism ring of the curve E.

According to During's theorem [8], $End(E)$ is either an order in an imaginary quadratic field or an order in a quaternion algebra over $\mathbb{Q}$. An elliptic curve E is called ordinary in the first case and supersingular in the second case. supersingular elliptic curves are an important family of elliptic curves in isogeny-based quantum cryptography due to the hardness of finding an isogeny between given supersingular curves. The most well-known family of these curves are as follows:

i) $y^2 + y = x^3 + b$ over $\mathbb{F}_{2^m}$, m odd.
ii) $y^2 = x^3 + ax$ over $\mathbb{F}_{p^m}$, where $p \equiv 3 \ (mod\ 4)$.
iii) $y^2 = x^3 + b$ over $\mathbb{F}_{p^m}$, where $p \equiv 2 \ (mod\ 3)$.

In the following, we consider the notion of Weil pairing on elliptic curves. Let $E[r] = \{P \in E(\bar{\mathbb{F}}_q) \mid rP = \mathcal{O}\}$ where $\bar{\mathbb{F}}_q$ is an algebraic closure of $\mathbb{F}_q$ and $\mu_r = \{g \in \bar{\mathbb{F}}_q \mid g^r = 1\}$. The Weil pairing e_r is a map

$$e_r : E[r] \times E[r] \longrightarrow \mu_r$$

with the following properties [15]:

(1) For $S_1, S_2, T \in E[r]$,

$$e_r(S_1 + S_2, T) = e_r(S_1, T)e_r(S_2, T),$$
$$e_r(T, S_1 + S_2) = e_r(T, S_1)e_r(T, S_2).$$

(2) for $S, T \in E[r]$,

$$e_r(P, P) = 1$$
$$e_r(S, T) = e_r(T, S)^{-1}$$

(3) There is an efficient algorithm to compute $e_r(P, Q)$ for every $P, Q \in E[r]$.

(4) For the isogeny $\phi : E_1 \to E_2$ and $P \in E_1[r]$ and $Q \in E_2[r]$, we have

$$e_r(P, \widehat{\phi}(Q)) = e_r(\phi(P), Q).$$

3. Certificateless Identification Scheme

In this section, we briefly present the structure of Certificateless Identification scheme (CLI), first introduced by Chin et al. in [6] and then we review the CLI scheme proposed in [6] on elliptic curves to see an example (ECLI).

3.0.1. **Framework of certificateless identification scheme.** Certificateless Identification scheme, includes six basic algorithms as follows:

Setup The KGC executes the **Setup** algorithm. The input of this algorithm is the security parameter 1^k and the output is the master secret key msk and master public key mpk. The KGC keeps secret msk and publishes mpk.

Partial-Private-Key-Extract The KGC runs this algorithm. The KGC takes the user's identity ID as input and generates partial private key ppk_{ID}. Then he sends the generated ppk_{ID} securely to the user.

Set-User-Key This algorithm is done by the user. It takes in the security parameter 1^k and the user's identity ID as input and generates user's secret value sv_{ID} and its corresponding public key upk_{ID}.

Set-Private-Key The user runs this algorithm. Using identity ID, public key upk_{ID}, the secret value sv_{ID} and partial private key ppk_{ID}, the algorithm returns the user's secret key usk.

Identification-Protocol: This protocol consists of two algorithms **Prover** and **Verifier**. It is a challenge-response interaction between prover and verifier to confirm the user's identity. Both algorithms take the values of upk_{ID}, mpk and prover's identity ID as input. The prover takes also usk as the user's private key and starts to perform the protocol as follows:

1) **Prover** sends the commitment to the **Verifier**.

2) **Verifier** sends the a value as the challenge to the **Prover**.

3) **Prover** sends the response to the **Verifier** and verifier will confirm the prover's identity if he could pass the verification test successfully.

3.1. **ECLI Scheme.** In [6], Chin et al. provide an efficient pairing-free CLI scheme which is also free from the key escrow and certificate management problem. In the following, we represent this scheme on the additive group of points of an elliptic curve over a finite field.

Setup (1^k)

1. For security parameter k and finite field $\mathbb{F}_q$, define an elliptic curve E over $\mathbb{F}_q$ and for the random element $a \in \mathbb{Z}_q^*$, set $P_{KGC} = aP$.

2. Select two hash functions $H_1 : \{0,1\}^* \times E(\mathbb{F}_q) \times E(\mathbb{F}_q) \to \mathbb{Z}_q^*$ and $H_2 : \{0,1\}^* \times E(\mathbb{F}_q) \times E(\mathbb{F}_q) \times E(\mathbb{F}_q) \to \mathbb{Z}_q^*$.

4. Publish the master public key $mpk = \{E, q, H_1, H_2, P_{KGC}, P\}$ and keep the master secret key $msk = a$.

Partial-Private-Key-Extract(mpk, msk, ID)

1. Select a random element $x \in \mathbb{Z}_q^*$ and compute $X = xP$,
2. Compute $\alpha = H_1(ID, P_{KGC}, X)$,
3. Compute $d = x - a\alpha$,
3. Return the partial private key $ppk = \langle \alpha, d \rangle$.

Set-User-Key(1^k)

1. Select a random $b \in \mathbb{Z}_q^*$ and set $s_{ID} = b$ as the secret value of the user,
2. Compute $Q = bP$,
3. Output $U_{pk1} = Q$.

Set-Private-Key$(mpk, ppk, s_{ID}, U_{pk}, ID)$

1. Calculate $X = dP + \alpha P_{KGC}$ and check if $\alpha = H_1(ID, P_{KGC}, X)$,
2. If correct, then calculate $\beta = H_2(ID, P_{KGC}, X, Q)$,
3. Compute $s_{ID} = d - b\beta$,
4. Compute $U_{pk2} = \beta Q$,
5. Publish $U_{pk} = \langle U_{pk1}, U_{pk2} \rangle$ and keep secret $U_{sk} = \langle \alpha, \beta, s_{ID} \rangle$.

Identification-Protocol: Prover (mpk, ID, U_{sk}) **and Verifier** (mpk, ID, U_{pk})

1. **Prover** selects a random $r \in \mathbb{Z}_q^*$ and compute $R = rP$.
2. **Prover** also computes $X = s_{ID}P + \alpha P_{KGC} + \beta Q$ and sends X, R to **Verifier**.
3. **Verifier** selects a random $c \in \mathbb{Z}_q^*$ and sends c to **Prover**.
4. **Prover** Computes response $y = r + cs_{ID}$ and sends y to **Verifier**.
5. **Verifier** accepts if and only if $yP = R + c(X - (\alpha P_{KGC} + \beta Q))$ and $U_{pk2} = \beta.U_{pk1}$, where $\alpha = H_1(ID, P_{KGC}, X)$ and $\beta = H_2(ID, P_{KGC}, X, Q)$.

To prove correctness, one can show that

$$\begin{aligned} yP &= (r + cS_{ID})P \\ &= rP + c(x - (a\alpha + b\beta))P \\ &= R + c(X - (\alpha P_{KGC} + \beta Q)) \end{aligned}$$

4. **Our New Scheme**

In this section, we introduce our certificateless scheme using isogenies between supersingular elliptic curves. The security of this scheme is based on the problem of finding isogenies between supersingular elliptic curves which is quantum-resistant [9] and some pairing problems. Then we generalize the proposed scheme such that the verifier can verify more than one identity by a single execution of the protocol.

Setup (1^k)

1. For security parameter k, KGC selects the finite field $\mathbb{F}_q$ and a supersingular elliptic curve E_1 over $\mathbb{F}_q$.

2. It selects a point $G \in E_1$ and computes an isogeny $\phi : E_1 \to E_2 = E_1/\langle G \rangle$. Then it selects a point $P \in E_2$ and sets $P_{KGC} = \widehat{\phi}(P) \in E_1$.

3. It selects a hash function $H : \{0,1\}^* \to E_1$.

4. It publishes the master public key $mpk = \{q, E_1, E_2, H, P_{KGC}, P\}$ and keeps the master secret key $msk = G$.

Partial-Private-Key-Extract (mpk, msk, ID)

1. KGC computes $Q_{ID} = H(ID) \in E_1$ and a positive integer n such that $P, Q_{ID} \in E_1[n]$ and $e_1(P_{KGC}, Q_{ID}) \neq 1$ where $e_1 : E_1[n] \times E_1[n] \to \mu_n$ be the Weil pairing defined over $E_1[n]$.

2. It computes $S_{ID} = \phi(Q_{ID})$.

3. It returns the partial private key $ppk = (S_{ID}, n)$.

Set-User-Key (1^k)

1. User selects a random $b \in \mathbb{Z}_n$, $P' \in E_2[n]$ and sets the user's secret value $s_{ID} = (b, P')$.

Set-Private-Key (mpk, ppk, s_{ID}, U_{pk}, ID)

1. User checks $e_2(P, S_{ID}) = e_1(P_{KGC}, Q_{ID})$ where $e_2 : E_2[n] \times E_2[n] \to \mu_n$ be the Weil pairing defined over $E_2[n]$.

2. If correct, then he selects a point P' randomly and computes an isogeny $\psi : E_2 \to E_3 = E_2/\langle bS_{ID} + P' \rangle$.

3. He computes $U_{pk} = \psi(bP)$. Then publishes U_{pk} and the Weil pairing $e_3 : E_3[n] \times E_3[n] \to \mu_n$.

4. He sets $U_{sk} = bS_{ID} + P'$.

Identification-Protocol: Prover (mpk, ID, U_{sk}) **and Verifier** $(mpk, ID, U_{pk}, P_{KGC})$

1. **Prover** selects a random point $R \in E_3$.

2. **Prover** computes the commitment $S = R + \psi(S_{ID})$ and sends (S, E_3, n) to **Verifier**.

3. **Verifier** checks if $e_3(U_{pk}, S) \neq 1$, then he selects a random $d \in \mathbb{Z}_n^*$ and sends d to **Prover**.

4. **Prover** computes the response $Y = d\ U_{sk} + b\widehat{\psi}(S) + S_{ID}$ and $W = e_2(P, U_{sk})$ and then sends (Y, W) to **Verifier**.

5. **Verifier** accepts if and only if $e_2(P, Y) = e_3(U_{pk}, S)e_1(P_{KGC}, Q_{ID})W^d$.

Remark 4.1. To impersonate user ID, the adversary should compute $Y = d\,U_{sk} + b\widehat{\psi}(S) + S_{ID}$ in the verification step. To compute this point, the adversary should know $U_{sk} = Kernel(\psi)$ and $S_{ID} = \phi(Q_{ID})$, which requires knowing ϕ and ψ.

Lemma 4.2. *The previous scheme has the completeness and soundness properties.*

Proof. We prove that a legitimate user can be verified by an honest verifier as follows.

$$\begin{aligned} e_2(P, Y) &= e_2(P, b\widehat{\psi}(S) + S_{ID} + d\ U_{sk}) \\ &= e_2(P, b\widehat{\psi}(S))e_2(P, S_{ID})e_2(P, bS_{ID} + P')^d \\ &= e_3(b\psi(P), S)e_1(\widehat{\phi}(P), Q_{ID})W^d \\ &= e_3(U_{pk}, S)e_1(P_{KGC}, Q_{ID})W^d. \end{aligned}$$

To prove the soundness property, we show that forging the user's identity at least twice implies that the cheater has some parts of the user's secret key or is able to compute them. We assume that the values (Y_1, d_1) and (Y_2, d_2) are true in the verification relation. Therefore $U_{sk} = (Y_2 - Y_1)(d_2 - d_1)^{-1}$ and cheater can have access to the value U_{sk}. □

4.1. **Batch Verification Scheme.** In this section, we show that we can extend our proposed protocol such that a user could be able to verify a bunch of identities instead of only once. This idea helps the user to efficiently execute the protocol only once to verify many identities at the same time. Assume that we have a bunch of identities $\{ID^1, \ldots, ID^k\}$, so the user obtains the corresponding private keys $\{S_{ID}^1 = \phi(Q_{ID}^1), \ldots, S_{ID}^k = \phi(Q_{ID}^k)\}$ from the KGC, where $Q_{ID}^i = H(ID^i)$ for $i = 1, \ldots, k$. Let $\bar{S}_{ID} = \sum_{i=1}^{k} S_{ID}^i$ and $\bar{Q}_{ID} = \sum_{i=1}^{k} Q_{ID}^i$. To verify the

private keys, the user checks if $e_2(P, \bar{S}_{ID}) = e_1(P_{KGC}, \bar{Q}_{ID})$. This equality holds because

$$\begin{aligned} e_2(P, \bar{S}_{ID}) &= e_2(P, \sum_{i=1}^{k} S_{ID}^i) \\ &= e_2(P, \sum_{i=1}^{k} \phi(Q_{ID}^i)) \\ &= e_2(P, \phi(\sum_{i=1}^{k} (Q_{ID}^i)) \\ &= e_1(\widehat{\phi}(P), \sum_{i=1}^{k} Q_{ID}^i) \\ &= e_1(P_{KGC}, \overline{Q}_{ID}). \end{aligned}$$

In the identification step, the prover selects $R \in E_3$ and sends $S = R + \psi(\bar{S}_{ID})$ to verifier. By receiving the value $d \in \mathbb{F}_q$ from the verifier, user sends the response $Y = d\, U_{sk} + b\widehat{\psi}(S) + S_{ID}$ and $W = e_2(P, U_{sk})$ to verifier. Then verifier checks if

$$e_2(P, Y) = e_3(U_{pk}, S) e_1(P_{KGC}, \bar{Q}_{ID}) W^d.$$

5. Efficiency and Security Analysis

In this section, we analyze the complexity and security of our proposed identification scheme and then we review the reset attack and its effect on our protocol.

In the following, we consider the required tools and their computational complexity to execute each step of our protocol and then we summarize the total number of needed operations in Table 5. One of the essential requirements of implementing the proposed protocol is constructing an isogeny when its kernel is available. As we see in the scheme, KGC needs to construct the isogeny ϕ and the user needs to construct the isogeny ψ in the Setup and Set-Private-Key algorithms respectively. The following procedure shows that how we can construct such isogenies in general.

Let E be an elliptic curve. Then for each subgroup G of E with $|G| = \ell$, there exists an isogeny $\phi : E \to E' = E/G$ (unique up to isomorphism) with $ker(\phi) = G$. In order to find such an isogeny, one can use the Velu's formula with the running time $O(\ell)$ [19]. Moreover, in the case where $|G| = \ell^e$ and ℓ is a small prime, one can use the Jao's efficient proposition [9] to compute ϕ as follows. Set $E_0 = E$, $G_0 = G$ and compute

$$E_{i+1} = E_i / \langle \ell^{e-i-1} G_i \rangle, \qquad \phi_i : E_i \to E_{i+1}, \qquad G_{i+1} = \phi_i(G_i)$$

for $i = 0, \ldots, e-1$. Then $E/\langle G \rangle = E_e$ and $\phi = \phi_{e-1} \circ \cdots \circ \phi_0$.

We also need to compute dual of isogenies during the execution of the protocol. Since $Ker(\widehat{\phi}) = \phi(E[l])$, one can compute the dual isogeny $\widehat{\phi}$ of an l-isogeny ϕ using the Velu's formula.

We also need to compute some pairings during the execution of the protocol. Miller's algorithm efficiently computes the pairing of two points of order n in $O(\log n)$ operations [15].

The following table shows the required operations to execute each step of the proposed protocol. We omit the field operations and also the addition of points on an elliptic curve due to their negligible complexity compared to other operations. The required operations consist of scalar Multiplication (M), Pairing (P), Hashing (H), Isogeny (I) and its Evaluation (E) computations.

TABLE 1. Operation costs for our protocol

Algorithm	M	H	I	P	E
Setup	0	0	1	0	1
Partial-Private-Key	0	1	0	0	1
Set-User-Key	0	0	0	0	0
Set-Private-Key	2	1	1	0	1
Prover	2	0	1	1	2
Verifier	0	0	0	3	0

One can use the following parameter selection to have a more efficient setting. Assume that KGC selects a finite field $\mathbb{F}_q$ where $q = p^{2k}$ and a supersingular elliptic curve E_1 over $\mathbb{F}_q$. Then by [21] we have:

$$E_1(\mathbb{F}_q) \cong \mathbb{Z}/(p^k + (-1)^{k+1})\mathbb{Z} \times \mathbb{Z}/(p^k + (-1)^{k+1})\mathbb{Z}.$$

Let P_1 and P_2 be two generators of E_1 with orders n_1 and n_2 respectively. KGC selects a point $G \in E_1$ of prime order ℓ and computes an isogeny $\phi : E_1 \to E_2 = E_1/\langle G\rangle$. Now by setting $P_{KGC} = \ell P_1$ and $P = \ell'\phi(P_{KGC})$, where ℓ' is the inverse of ℓ in $\mathbb{Z}/n_1\mathbb{Z}$, we have

$$\widehat{\phi}(P) = \ell'\widehat{\phi}(\phi(P)) = \ell'\ell P_{KGC} = P_{KGC}.$$

Finally, we can choose $H : \{0,1\}^* \to \mathbb{Z}_{n_2}$ and set $Q_{ID} = H(ID)P_2$ to avoid trivial pairing evaluations.

Besides some pairing problems, the security of our protocol is based on the hardness of isogeny problem:

Isogeny Problem: Let E_1 and E_2 be two isogenous elliptic curves. Find an isogeny $\phi : E_1 \longrightarrow E_2$.

There have been many efforts to attack the isogeny problem in the literature. In 2013, Galbraith and Stolbunov introduced an algorithm which solves the isogeny problem for ordinary curves over finite field $\mathbb{F}_q$ in $\widetilde{O}(q^{1/4})$, where $\widetilde{O}$ denotes the complexity with the logarithmic factors omitted [10]. For supersingular elliptic curves, one can use Delfs and Galbraith's classical algorithm which solves the isogeny problem in $\widetilde{O}(p^{1/2})$ operations, where p is the characteristic of the base field [7].

In quantum computing, there exists a subexponential quantum algorithm that breaks the isogeny problem for ordinary elliptic curves using the commutativity of the endomorphism rings of these curves. In contrast, supersingular curves are secure against this attack due to the non-commutative property of their corresponding endomorphism ring. As the best-known quantum algorithm for attacking to supersingular elliptic curves is proposed by Biasse et al. [3] which solves the isogeny problem with exponential running time $\widetilde{O}(p^{1/4})$, supersingular elliptic curves seem to be promising candidates for post-quantum cryptography.

Finally, we consider reset attack in which the cheating verifier can reset the internal state of the prover to obtain some secret information.

Lemma 5.1. *In our proposed scheme, if the prover P uses the same commitment S within two subsequent conversations in the identification step, then the verifier V can have access to a part of the user's secret information.*

Proof. Suppose that prover P chooses the same R in two different conversation with the verifier V. Suppose also that the cheater verifier V intentionally sends the challenges d and $d+1$ to P and receives two corresponding responses Y_1 and Y_2. Then the cheater verifier obtains the user's secret value $U_{sk} = Y_2 - Y_1$. □

Therefore, according to the protocol, it is sufficient that the prover P selects R randomly in commitment stage to prevent this attack.

6. EXAMPLE

In this section, we present an example of the implementation of the scheme for a 180-bit finite field. This example is executed by the SAGE software [16, 17]. The example goes as follows:

Setup

$p = 490015260127433451783546712934103296816 9,$
$\mathbb{F}_q = \frac{\mathbb{F}_p[a]}{\langle a^2+a+1\rangle},$
$E_1 : y^2 = x^3 + 1$ over $\mathbb{F}_q,$

$$P_1 = (P_{1,x}, P_{1,y}),$$

$$P_{1,x} = 49348603615790565796427516146943105271 2a + 437379845954127808652829842057997361666 3$$

$$P_{1,y} = 12855858845986961513760658676911095716 58a + 23927407592966566982664377342445395334 44.$$

$$P_2 = (P_{2,x}, P_{2,y}),$$

$$P_{2,x} = 12754549734846393937022597522328063516 3a + 28809705683675099966276799143449582891 75$$

$$P_{2,y} = 45230702275480638468801896017480091022 3a + 63256888007395730181771965523281344249 1.$$

$$n_1 = n_2 = 490015260127433451783546712934103296817 0.$$

$$G = (g_x, g_y) \in E_1,$$

$$g_x = 85459118173112773930154918779946664194 1a + 11532216158531532785673033133315013948 97$$

$$g_y = 36145366906308905428416918806740356211 57a + 14613457377378562561229504799581150409 11.$$

$\phi : E_1 \to E_2 : y^2 = x^3 + a_2x + b_2$ is an 1051-isogeny, where

$$a_2 = 42469268529293948999434979011548309467 94a + 19086130764569247255802771657793388903 7,$$

$$b_2 = 27052322555980768763505147756837624468 00a + 65469823215781857341198375594010529121 0.$$

$$P_{KGC} = (P_{KGC,x}, P_{KGC,y}) \in E_1$$

$$P_{KGC,x} = 15971503713847379570752281454361521640 13a + 39010159528502921374537028606143930371 16,$$

$$P_{KGC,y} = 23681479360200297851099664385694094609 06a + 22191768162338169564934530702984021304 67.$$

$$\ell' = 22979148556448032187538957261478625 11$$
$$P = (P_x, P_y) \in E_2,$$

$$P_x = 41302923243319542006555610672561046596 42a + 21238417384222711434037109612573390004 95,$$

$$P_y = 99838808841438741875730876734612940206 1a + 19637263510276327866182872535775440660 0.$$

$mpk = \{q, E_1, E_2, H, P_{KGC}, P\}$ where $msk = G$ and $H : \{0,1\}^* \longrightarrow \mathbb{Z}_{n_2}$

Partial-Private-Key-Extract(mpk, msk, ID)

$H(ID) =$133197711706781

$$Q_{ID} = (Q_x, Q_y) \in E_1,$$

$$Q_x = 35663641123420964420909054757749792219 42a + 28968305495268012304998901387399845 1346,$$

$$Q_y = 33917948033868486193679550819764257373 28a + 77529223403769199567461890330363672426.$$

$$m = n = 46623716472638768009852208652150 64670.$$

$$S_{ID} = (S_{ID,x}, S_{ID,y}),$$

$$S_{ID,x} = 21132802603988818470079314582113569557 2a + 28561261331787069089607227513997492055 6,$$

$$S_{ID,y} = 41652261731886628423476548316269929778 48a + 23516796075955743576104966638272883314 19.$$

Set-User-Key

$$b = 41740122177832379597002872562355 10 \in \mathbb{Z}_q.$$

Set-Private-Key $(mpk, ppk, s_{ID}, U_{pk}, ID)$

$$e_2(P, S_{ID}) = e_1(P_{KGC}, Q_{ID}) = c_1,$$

$$c_1 = 23720517641850459112204857773371406944 28a + 47335487490913855831544818253194808805 61$$

$$P' = (P'_x, P'_y) \in E_3,$$

$$P'_x = 15977408067936182776373578607359232722 70a + 47380380780057628570151985336034365240 63,$$

$$P'y = 14719045658365230432273879918299104823 11a + 29402488911321793761231539421841096549 79.$$

$$Usk = (U_{sk,x}, U_{sk,y}),$$

$$Usk_x = 54991665331281254633659958596856564325 33a + 20304684156303264402352226079569818071 22,$$

$$Usk_y = 19440550898006310882420677439507246817 79a + 18579409269383636413441640020073961618 01.$$

The 1117-isogeny $\psi : E_2 \to E_3 = y^2 = x^3 + a_3x + b_3$,

$$a_3 = 42469268529293948999434979011548309467 94a + 19086130764569247255802771657793388903 7,$$

$$b_3 = 27052322555980768763505147756837624468 00a + 65469823215781857341198375594010529121 0.$$

$$U_{pk} = (U_{pk,x}, U_{pk,y}),$$

$$U_{pk,x} = 17398819045422485567201980019005158798 89a + 47186043477166813648148349445924086910 10,$$

$$U_{pk,y} = 26612714419617129766012182124566974721 1a + 14962475179944921466153381522935123265 33.$$

Identification-Protocol:

Prover (mpk, ID, U_{sk}) **and Verifier** $(mpk, ID, U_{pk}, P_{KGC})$

Prover:

$R = (R_x, R_y) \in E_3$,

$$R_x = 468262535396234935177192529132482467542 3a + 104048795197012530368151669250795662 2982,$$

$$R_y = 165164078362719010149098686283125291682 8a + 342764206711935281265998959395086446774.$$

$S = (S_x, S_y)$,

$$S_x = 450278451073695377547792898894561340857 0a + 193767173718127073479939867303661350578,$$

$$S_y = 259733418289169445171600933190918780048 3a + 591711528527138120018733168402330021810.$$

Verifier:

$d = 15747947806697752759 \in \mathbb{F}_q$.

Prover:

$Y = (Y_x, Y_y)$ and W,

$$Y_x = 482076240368979981028646210883603455639 4a + 104085006082898050788745991838262469827 5,$$

$$Y_y = 110686938856545878378725390177479006037 5a + 424760200593982056027394462581713813455 0,$$

$$W = 107674933081889429905480031248248128968 7a + 234779890199496735386509310951054530966 7.$$

Verifier:

$e_2(P, Y) = e_3(U_{pk}, S)e_1(P_{KGC}, Q_{ID})^d = c_2$,

$$c_2 = 434723137622647304786417078355088333502 3a + 353298735412237934312602286604341054396 5.$$

7. Acknowledgments

This research was in part supported by the University of Kashan under grant number 159037/1.

References

[1] S. S Al-Riyami and K. G Paterson. Certificateless public key cryptography. *Asiacrypt*, 2894, (2003), 452-473.

[2] M. Bellare, Ch. Namprempre, and G. Neven, Security proofs for identity-based identification and signature schemes. *Journal of Cryptology*, 22(1),(2009),1-61.

[3] J.F. Biasse, D. Jao, and A. Sankar, A quantum algorithm for computing isogenies between supersingular elliptic curves. *International Conference in Cryptology in India*, (2014), 428-442.

[4] A. Childs, D. Jao, and V. Soukharev, Constructing elliptic curve isogenies in quantum subexponential time. *Journal of Mathematical Cryptology*, 8(1),(2014),1-29.

[5] J.J. Chin, R.C-W Phan, R. Behnia, and S.H. Heng, An efficient and provably secure certificateless identification scheme. *Security and Cryptography (SECRYPT), 2013 International Conference on*, (2013), 1-8.

[6] J.J. Chin, S.Y. Tan, S.H. Heng, R.C-W Phan, and R. Behnia, A provable secure pairing-free certificateless identification scheme. *International Journal of Computer Mathematics*, 92(8),(2015),1520-1535.

[7] Ch. Delfs and S.D. Galbraith, Computing isogenies between supersingular elliptic curves over $\mathbb{F}_p$ Designs, Codes and Cryptography, 78(2),(2016), 425-440.

[8] M. Deuring. Die typen der multiplikatorenringe elliptischer funktionenkrper. *Abhandlungen aus dem mathematischen Seminar der Universitt Hamburg*, 14, (1941), 197-272.

[9] D. Feo, Luca, Jao, David and Plt, Jrme Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies *J. Math. Cryptology*, 8(3), (2011), 209-247.

[10] S. Galbraith and A. Stolbunov, Improved algorithm for the isogeny problem for ordinary elliptic curves. *Applicable Algebra in Engineering, Communication and Computing*, 24(2), (2013), 107-131.

[11] D.R. Kohel, *Endomorphism rings of elliptic curves over finite fields*. Thesis, 1996.

[12] K. Kurosawa and S.H. Heng, From digital signature to id-based identification/signature. *International Workshop on Public Key Cryptography*, (2004), 248-261.

[13] R.L. Rivest, A. Shamir, and L. Adleman, A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 26(1), (1983), 96-99.

[14] A. Shamir, Identity-based cryptosystems and signature schemes. *Crypto*, 84, (1984), 47-53.

[15] J.H. Silverman, *T*he arithmetic of elliptic curves, volume 106. Springer Science & Business Media, 2009.

[16] W.A. Stein, and D. Joyner, SAGE: System for Algebra and Geometry Experimentation, Comm. Computer Algebra 39 (2005), 61–64.

[17] W.A. Stein, SAGE: Software for Algebra and Geometry Experimentation, http://www.sagemath.org.

[18] J. Tate, Endomorphisms of abelian varieties over finite fields. *Inventiones mathematicae*, 2(2), (1966), 134-144.

[19] J. Vlu, Isognies entre courbes elliptiques. *Comptes-Rendus de l'Acadmie des Sciences*, 273, (1971),238-241.

[20] L.C. Washington, *E*lliptic curves: number theory and cryptography. CRC press, 2008.

[21] Ch. Wittmann, Group structure of elliptic curves over finite fields. *Journal of Number Theory*, 88(2), (2001), 335-344.

Hassan Daghigh

Faculty of mathematical sciences, University of Kashan, P.O.Box 8731751167, Kashan, Iran.

hassan@kashanu.ac.ir

Ruholla Khodakaramian Gilan

Faculty of mathematical sciences, University of Kashan, P.O.Box 8731751167, Kashan, Iran.

rkhodakaramian@grad.kashanu.ac.ir